NASA TECHNICAL TRANSLATION NASA TT F-13,872

REDUCING THE STRUCTURAL AND FUNCTIONAL REDUNDANCY IN COMPLEX INFORMATION SYSTEMS

Ya. A. Dubrov

N72-14174 (NASA-TT-F-13872) REDUCING THE STRUCTURAL AND FUNCTIONAL REDUNDANCY IN COMPLEX INFORMATION SYSTEMS Y.A. Dubrov
Unclas (Scientific Translation Service) Aug. 1971
12022 10 p CSCL 09D G3/07

Translation of "Sokrashcheniye struyktyrnoy i funktsional'noy izbytochnosti v slozhnykh informatsionnykh sistemakh". Chetverty simpozium po probleme izbytochnosti v informatsionnykh sistemakh. (Fourth Symposium on the Problem of Redundancy in Information Systems), Leningrad, USSR Academy of Sciences, 1970, pp 216-223

FACILITY FORM 602
(ACCESSION NUMBER) 10
(PAGES) 1
(NASA CR OR TMX OR AD NUMBER)
(THRU) G3
(CODE) 02
(CATEGORY)

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON, D.C. 20546 AUGUST 1971

REDUCING THE STRUCTURAL AND FUNCTIONAL REDUNDANCY IN COMPLEX INFORMATION SYSTEMS

/216*

Ya. A. Dubrov

ABSTRACT. A finite system, developed by the author, is used to study some problems of complex information systems. Significant attention is devoted to questions of equivalent system transformations, since such transformations make it possible to eliminate structural and functional redundancy in complex information systems.

1. The conception of a system. Every system is a combination of a structure and a function. /217

Definition 1. A structure is defined as an oriented logic diagram (dimultigraph) with loops and colored lines F. It contains sets of nodes X and sets of lines U. These sets are ordered (by number, for example). Each of the lines $u \in U$ connects a pair of nodes, one of which may be empty.

An empty node is called a terminal. If the initial node of a line is empty, it is called an input line; if the final node of a line is empty, it is called an output line.

Definition 2. The function of a system is defined as the mapping (morphism) of a set of input signals of the system into a set of output signals.

---

*Numbers in the margin indicate the pagination in the original foreign text.

Definition 3. A system is defined as a structure in which every node corresponds to a function, and every line corresponds to a signal (an input signal for an input line and an output signal for the output).

2. System connections. System algebra. Three types of connections can exist between two systems. These connections lead to three types of actions of one system on another: a) The systems do not interact. b) The first system affects the second, but the second does not affect the first. c) Each of the systems affects the other.

These three types of connections logically lead to three operations on the set of systems: direct summation, multiplication, and iteration [1-2].

Definition 4. The direct sum F of the structures F_1 and F_2 is defined as the binary operation $F = F_1 + F_2$, which places the set $F = F_1 \cup F_2$ in correspondence with the sets F_1 and F_2. Set $F = F_1 \cup F_2$ is so ordered, so that $p \leq q$ in $F_1 \times F_1$ (or in $F_2 \times F_2$) for each pair (p,q) from $F_1 \times F_2$, for no pair from /218 $F_2 \times F_1$, and for the pairs from $F_1 \times F_1$ (or from $F_2 \times F_2$).

Definition 5. The product F of the structures F_1 and F_2 is defined by the binary operation, which places the structure $F = F_1 \circ F_2$ in correspondence to each order pair of the structures $\langle F_1, F_2 \rangle$, according to the following rule: At least one output of the structure F_1 is connected with at least one input of the structure F_2; the inputs of the structure F are the inputs to the structure F_1 and those inputs to F_2 which are not connected to the outputs of F_1; the outputs of the structure F are the outputs of the structure F_2 and those F_1 outputs which are not connected to the inputs of the structure F_1.

Definition 6. The iteration of the structure F is defined as the unary operation $\{F\} = F \cdot F$, such that at least one of the outputs of the structure F is connected with at least one of its inputs.

Definition 7. The structure algebra is defined as the set of structures with their given operations of direct summation, multiplication, and iteration.

Definition 8. The direct summation of functions is defined as the binary operation which places the function-sequence $\vec{f}(x_1,\ldots,x_n) = f_1(x_1,\ldots,x_m) + f_2(x_{m+1},\ldots,x_n) = \langle f_1(x_1,\ldots,x_m), f_2(x_{m+1},\ldots,x_n)\rangle$ in correspondence with the functions $f_1(x_1,\ldots,x_m)$, and $f_2(x_{m+1},\ldots,x_n)$. The first component of $\vec{f}$ is f_1, and the last component is f_2.

Definition 9. The superposition of the functions $f_1(x_1,\ldots,x_{i-1},x_i,x_{i+1},\ldots,x_m)$ and $f_2(y_1,\ldots,y_n)$ is defined as the binary operation whose result is a function f, which is the same as the function f_1 with one of its arguments replaced by the function f_2. Thus, we have $f(x_1,\ldots,x_{i-1},y_1,\ldots,y_n, x_{i+1}, x_m) = f_1(x_1,\ldots, x_{i-1},\cdot, x_{i+1},\ldots,x_m)\cdot f_2(y_1,\ldots,y_n) = f_1(x_1,\ldots x_{i-1}, f_2(y_1,\ldots,y_n), x_{i+1},\ldots,x_m)$.

Definition 10. The iteration of the function $f(x_1,\ldots,x_{i-1}, x_i, x_{i+1},\ldots,x_n)$ on the argument x_i is defined as the unary operation: /219

$$\{f(x_i,\ldots,x_{i-1}, x_i, x_{i+1},\ldots,x_n)\}_0 = \ldots\ldots\ldots\ldots\ldots\ldots$$

$$\{f(x_1,\ldots,x_{i-1}, , x_{i+1},\ldots,x_n)\}_k = f(x_1,\ldots, x_{i-1}, \{f(x_1,\ldots, x_{i-1},\cdot,$$

$$x_{i+1},\ldots,x_n)\}_{k-1}, x_{i+1},\ldots,x_n), \ldots\ldots\ldots\ldots\ldots\ldots$$

where $\{f(x_1,\ldots,x_n)\}_k$ is the k^{th} iteration of the function $f(x_1,\ldots,x_n)$.

Definition 11. The set of functions and the summation superposition, and iteration operations which are defined for them are called the function algebra.

We now construct the system algebra, and introduce the following three operations on the system set: direct summation, multiplication, and iteration.

Definition 12. The direct sum A of the systems A_1 and A_2 is defined as the binary operation that produces the system $A = A_1 + A_2$, such that the

structure of A is the direct sum of the structures of the components, and the corresponding function is the direct sum of the functions of the system components.

Definition 13. The product A of the systems A_1 and A_2 is defined as the binary operation which places the system $A = A_2 \cdot A_1$ in correspondence with each pair of the systems, such that the structure of A is equal to the product of the structures of these systems, and the function of A is equal to the superposition of the functions of the system components.

Definition 14. The iteration B of the system A is a unary operation, which places the system $B = \{A\}$ in correspondence with each system, such that the structure of B is equal to the iteration of the structure of the system A, and the function of B is equal to the iteration of the functions of the system A.

Definition 15. The system algebra is defined as the set of systems and the operations of direct summation, multiplication, and iteration, which are defined on these systems.

3. Some remarks and limitations. The system will be expressed in terms /220
of a flow diagram. Aside of each node in the system, we will write the corresponding function along with its dimensions [$f(\cdot,\cdot)$, for example]. If several lines lead from one node, it is necessary to use arrows to indicate that the function is a vector function. We will write the output signal at the output lines of a node, when there is more than one output line, and the output signals from the node are different from each other. We assign input and output signals to every one of the corresponding system inputs and outputs. A number is assigned to each of the multiple input and output lines as well as to each node. If there are feedback loops for some nodes, that is, if there is an iteration operation on the corresponding system, it is necessary to use the

same color for the feedback line and for the line which is assigned to the iteration variable[1].

Generally speaking, systems may have an arbitrary number of inputs and outputs, but we will not consider those systems without inputs (exclusively active systems), without outputs (exclusively passive systems), or without inputs or outputs (absolutely isolated systems or monads). We shall also assume that the number of inputs and outputs of the system is finite. Thus, if *(S) designates the set of inputs to the system S, and E(S) designates the set of outputs, then for the systems being considered:

$$1 \leqslant \operatorname{card} I(S) < \infty,$$
$$1 \leqslant \operatorname{card} E(S) < \infty,$$

where "card A" designates the cardinal number of the set A. /221

System functions can be undefined for some values of the function arguments. Systems with such functions and the functions themselves will be called partially defined. A function of a complex system is undefined, if one of the functions of the elementary systems that comprise the complex system is undefined.

We note that our definition of a function can include operators, relations, various functions in the narrow meaning of this work, algorithms, programs, automatons, conditional probabilities, etc. In this connection, it is possible to consider operator systems, algorithmic systems, automaton systems, probability systems, and other systems.

4. Equivalence, isomorphism, identity.

---

[1] We will use different types of arrows in the figures instead of different colors.

Definition 16. Systems, which have isomorphic structures, are called isomorphic systems.

Definition 17. Structures are called isomorphic if it is possible to establish mutually identical relationships between their elements (lines and nodes).

Definition 18. Systems are called identical if they have identically equal functions.

Equivalent systems are the same from the point of view of external experiments.

Definition 19. Systems, which are both isomorphic and equivalent, are called identically equal or identical.

Definition 20. The transition from one system to an equivalent system is called an equivalent transformation.

The study of equivalent transformations is important in the study of system theory. The problem of describing all possible equivalent transformations and the problem of finding the complete system of equivalent relations /222 are particularly important. On the other hand, equivalent transformations make it possible to transform a system into an equivalent optimum system, which is important in synthesizing systems. The problem of studying algorithms for "simplifying" systems by means of equivalent transformations is particularly important [1-2]. These algorithms remove functional and structural redundancy in systems.

5. Equivalent transformations and redundancy. We now consider several examples of equivalent system transformations. The first of these simplifies

the system structure; the second simplifies the controlling formulations[2] which express the system functions. Here, simplification actually means the reduction of structural and functional redundancy in the system. The equivalent transformations which are given below are related to the iteration operation on one argument.

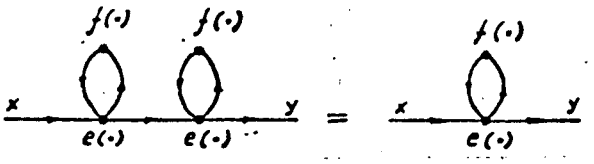

Figure 1. Multiplication of iterations.

Figure 1 shows an equivalent transformation, which can be written in the following manner:

$$y = \{f(\cdot)\} \cdot e(\cdot) \cdot \{f(\cdot)\} \cdot e(x) = \{f(\cdot)\} \cdot e(x).$$

that is,

$$\{f(\cdot)\} \cdot \{f(\cdot)\} = \{f(\cdot)\},$$

where e(x) = x is the identity function.

For Figure 2 we have /223

$$f(\cdot) \cdot \{f(\cdot)\} \cdot e(x) = \{f(\cdot)\} \cdot e(\cdot) \cdot f(x) = \{f(\cdot)\} \cdot f(x),$$

---

(2) By controlling formulations we mean formulations of system functions which are constructed by a finite number of direct summation, multiplication, and iteration operations.

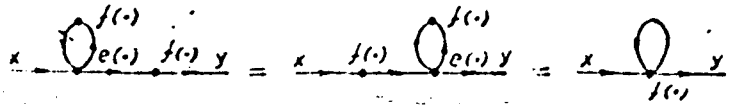

Figure 2. Commutation property of iterations.

that is,

$$\{f(\cdot)\} \circ f(\cdot) = f(\cdot) \circ \{f(\cdot)\}.$$

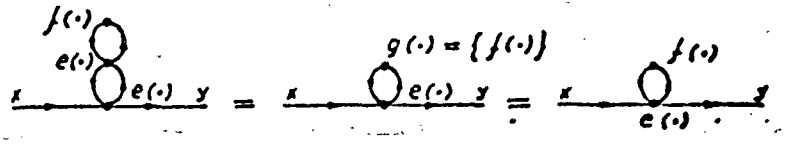

Figure 3. Idempotent iterations.

For Figure 3 we have

$$y = \{g(\cdot)\} \circ e(x) = \{\{f(\cdot)\}\} \circ e(x)$$

but on the other hand

$$\{g(\cdot)\}_0 = \cdot \ ,$$
$$\{g(\cdot)\}_1 = g(\cdot) = \{f(\cdot)\} \ ,$$
$$\{g(\cdot)\}_2 = g(\cdot) \circ g(\cdot) = \{f(\cdot)\} \circ \{f(\cdot)\} = \{f(\cdot)\}.$$

Thus, we have

$$\{\{f(\cdot)\}\} = \{f(\cdot)\}.$$

Besides the equivalent transformations discussed here, there are many others. Thus, one of the most important problems in system theory is finding some complete system of equivalent transformations, which could be the basis for constructing algorithms to reduce the structural and functional redundancy of complex systems (algorithms for minimizing systems).

REFERENCES

1. Dubrov, Ya.A. On the Theory of System Control. Doklad na nauchnoy konferentsii "Matematicheskiye metody v upravlenii proizvodstvom" (Report on the Scientific Conference "Mathematical Methods in Industrial Control"). Lvov, (in press), 1969.

2. Dubrov, Ya.O. The Algebra of I II. Vector Systems. Doklad na IV respublikanskoy konferentsii molodykh issledovateley "Teoreticheskiye i prikladnyye voprosy sistemotekhniki" (Report of the Fourth Republic Conference of Young Researchers "Theoretical and Applied Problems of System Technology"). Kiev, 1969.

Translated for NASA-Goddard Space Flight Center under Contract No. NASw 2035, by SCITRAN, P.O. Box 5456, Santa Barbara, California, 93108.